OAK LEAVES

ATOM USERS' GROUP CANADA
Newsletter #7

Well, here it is at last. We hope nobody had given up waiting. Gladstone may have given up as distributers for the Atom but the Users Group is not dead. Far from it! In this issue you may read of some of the developments which are currently very active.

The A.U.G.C. has maintained complete independance and never accepted paid advertisements. As commercial support for the Atom has dwindled to almost zero we have changed this policy slightly. There are still no paid ads. but in the interest of helping the members locate items of interest we will try to point out useful sources.

We have negotiated a discount for the Group when buying from Parts Galore at 331 Queen Street West, Toronto. (416) 598-0752. Albert will knock 5-10% off all purchases if you tell him you are a member of the Atom Users' Group. With this discount you can get a Shugart SA-400L for $200, a half height SA-200L for $165 or a Matsushita half height double sided drive for $230. These are bare drives. Add $10 if you want a case. They will ship anywhere in Canada (you pay the charges).

Nearly all members receiving this issue of Oak Leaves will get another issue before their membership expires. There is no need to renew until you are advised to do so.

Most back issues are still available. Send $3 for single issues or $8 for issues one through four.

John Wood
September 1984

This Newsletter is published by Atom Users' Group Canada, 812 Cabot Trail, Milton, ONT. L9T 3M8 for the benefit of members of the Group.

Many thanks to all those who have sent in contributions to Oak Leaves. Special thanks to Colin Hinz in Saskatoon, John Lasruk and Rick Bales in Toronto, and Michel Charest in Montreal.

STOP PRESS
There is apparently a bug in the UPDATE utility program mentioned in the Wordpak Plus article. Rick is looking into it. You are advised not to use it until the bug is found.

TESTING FOR THE TRUTH IN THE ATOM
by John Lasruk

"Awright, you punk computer! Are you gonna tell me the truth, or am I gonna hafta get rough?" I snarled.

Atom was a tough cookie, though, real tough. "Ya don't scare me, programmer," he spat, "I've been poked by better than you!"

"Oh, yeah? Well, how's about a little power surge; maybe that'd singe your synapses!!"

"Alright! Okay! I'll tell ya, I'll tell ya! The truth is..." I leaned forward in hungry anticipation, "...the truth is...NON ZERO!"

Well, sure it's silly, but the fact of the matter is, computers spend great chunks of time looking for the truth (or the non-truth). Any kind of comparison operation involves such a search, and since, even at machine level such commands as CMP and BNE are comparisons, there is a lot of questioning going on in there.

Naturally, the Truth, as we and the Good Book know it, is not the truth, as Atom knows it. When the creators of the Atom sat down to program, they knew that a whole lot of comparin' was goin' down and, in effect, had to assign a number each to: "Yes, this is so" (1) and, "No, this isn't" (0) Of course, "yes" and "no" are things that computers do very well. "Maybe" is where we have it all over them.

To get down to cases: Atom regards 0 (zero) as a falsehood and all other numbers as the truth. Not that he (she?) is incapable of dealing with zero; it's just that 0 is Atom's word for "NO!" A line in one of our programs might read, 120 IF $A="KOWABONGA!" PRINT "TARZAN EATS MONKEY MEAT.". Atom will do a bit-by-bit comparison of the contents of string A and the contents of the target string, let's call it the KOWABONGA string, merrily saying "1" for every comparison that matches, but shouting, "0" when a non-match pops up. The number chosen, be it a one or a zero, enables Atom to make the proper selection among the choices the programmer has supplied.

All of which is well and good, but I can hear you ask, "What flippin' good does it do for the BIG number 1...ME?" (You'll have to trust that I hear this.) Well, some of this "truthiness" manages to push its way up through the lower levels into our BASIC. For example, a perfectly valid substitute for "IF X<>0;Y=10" would be, "IF X;Y=10". Yes, it does look loony, but the IF statement looks around for some kind of truth and when X contains anything but zero, it is labelled "TRUE" and the rest of the statement is executed! (Which means, in this case, assigning a value of 10 to Y) The latter form happens to be the faster way to execute such an operation since it is shorter.

Another way to use truth and non-truth in your programs is to use the actual values that the Atom itself uses. A line like this: "X=X+1-(X=64)" looks like so much gobbdegook. I mean, how do you SUBTRACT a COMPARISON? Well, we're dealing with computers here, and computers don't really know about comparisons, just "yes" and "no", or, to be more specific, 1 and 0. If, in the example above, the variable X is actually equal to 64, Atom will assign the value 1 to the bracketed statement. If, on the other hand, the variable X is not equal to 64, the statement in the brackets is false and is assigned a 0 for its pains. The full equation, "X=X+1-(X=64)" may mean, according to the value of X, either "X=X+1-1", or, "X=X+1-0". In other words, the value of X remains the same or is incremented, depending on whether or not X was equal to 64 to start with.

This bit of trickery comes in especially useful in certain BASIC programs, particularly games, where a whole slew of IF statements can really slow the whole doings down to molasses city. The following little program does nothing but bounce a pixel around gr.0. You will note, however, there is nary an IF in sight. You may also notice that I threw a WAIT in there. In theory, this is to leave the pixel on the screen long enough to be seen. Actually, using graphics statements for motion is not the way to go; the program seems to run too fast for the graphics routines to catch up. Try deleting it and see what I mean.

```
100 CLEAR0
110 MOVE0,0;DRAW0,47      draw a box
120 DRAW63,47;DRAW63,0
130 DRAW0,0
140 X=32;Y=28             set start position of pixel
150 N=1;M=1
160aPLOT13,X,Y
180 WAIT                  things move too fast,otherwise
200 PLOT15,X,Y
210 GOSUBb;GOTOa
280bN=N+2*((Y=1)-(Y=46))  N may be 1 or -1
290 M=M+2*((X=1)-(X=62))  M may be 1 or -1
300 X=X+M;Y=Y+N
310 RETURN
```

Let's look at the statement in line 280. If Y=1, the first bracketed statement is true, therefore it is assigned the value 1; the second bracketed statement is false and assigned a 0. The entire equation then reads as follows: N=N+2*((1)-(0)), or, N=N+2. On the other hand, if Y=46, the first bracketed statement is false, the second is true, and the equation reads N=N+2*((0)-(1)), which is, N=N-2. If Y is neither 1 nor 46, the equation will read N=N+2*((0)-(0)), or, N=N+0 (no change). Though the line seems complex, it reads and is executed more quickly than the conventional IF statements it replaces.

Tech-Hint

Terry Ussher noticed a jumping of the 5 volt line when the disk was operating. He modified the original 9V power supply to give a regulated 5V and uses this to power the Atom board leaving the Disc Pack psu to power the disk drive. The result is a rock-solid line to the computer, and a much cooler running drive.

Interesting Addresses
The Saga Continues

1. Zero Page Addresses.

#00 - #5E	Utilized by BASIC
#5A - #61	Used by graphics routines
#64 - #6A	Used by Assembler
#5E - #7F	Used by Floating - Point Commands
#C0 - #C5	Used by the COS
#C9 - #DD	" " " "
#DE - #E7	Used by the Operating System
#E8 - #FC	Used by the COS
#FE - #FF	Used by the Operating System

2. DOS Usage of Zero Page RAM

#9A - #A3	Used for file arguements.
#A4	Misc. Use
#A5 - #AB	Used to hold filenames
#AC	Current Qualifier
#AD	Misc. Use
#AE - #B8	Used by *CAT only.
#B9 - #BF	Used by sequential files, plus *SPOOL and *EXEC
#C0	Misc. Use
#C1 - #C6	Used by sequential files, plus *SPOOL and *EXEC
#C7 - #CC	Misc. Use
#CD	Old Qualifier
#CE - #D4	Used by *VDU only.
#D5 - #D6	Misc. Use
#D7 - #DD	Free
#E2 - #E5	Used by *VDU only.
#E8	Free
#EE	Drive # (0-3)
#E9 - #FC	Misc. Use
#FD	Free

NOTE: Some other memory locations are used by the utility programs.

3. O.S. Vectors

Location	Name	Normally set to	Set by DOS to	Function
#200,#201	NMIVEC	not set	#E87B	For NMI Interrupts
#202,#203	BRKVEC	#C9D8	not set	For BREAK errors
#204,#205	IRQVEC	#A000	not set	For maskable interrupts
#206,#207	COMVEC	#F8EF	#E3E5	Command Line Interpreter
#208,#209	WRCVEC	#FE52	not set	Write character
#20A,#20B	RDCVEC	#FE94	not set	Read Character
#20C,#20D	LODVEC	#F96E	#E477	Load File
#20E,#20F	SAVVEC	#FAE5	#E613	Save File
#210,#211	RDRVEC	error	#EA61	Read File's Arguements
#212,#213	STRVEC	error	#ECA0	Store File's Arguements

#214,#215	BGTVEC	#FBEE	#EAF0	Get byte from tape/disk
#216,#217	BPTVEC	#FC7C	#EBBC	Put byte to tape/disk
#218,#219	FNDVEC	#FC38	#E953	Find file in DOS Print Messages in COS
#21A,#21B	SHTVEC	#C278	#E89E	Shut files in DOS

4.Remaining Block 0 Workspace

#21C - #23F	Free	
#240 - #24A	Variable used in For-Next statement	
#24B - #255	STEP values for FOR - NEXT LOOPS	(LOW BYTE)
#256 - #260	" " " " " "	
#261 - #26B	" " " " " "	
#26C - #276	" " " " " "	(HIGH BYTE)
#277 - #281	Upper bound of FOR - NEXT Loops	(LOW BYTE)
#282 - #28C	" " " " " "	
#28D - #297	" " " " " "	
#298 - #2A2	" " " " " "	(HIGH BYTE)
#2A3 - #2AD	FOR - NEXT stack (low bytes)	
#2AE - #2B8	FOR - NEXT stack (high bytes)	
#2B9 - #2C3	DO - UNTIL stack (low bytes)	
#2C4 - #2CE	DO - UNTIL stack (high bytes)	
#2CF - #2DC	GOSUB - RETURN stack (low bytes)	
#2DD - #2EA	GOSUB - RETURN stack (high bytes)	
#2EB - #305	Array Pointers (@@ in #2EB, ZZ in #305,etc. LOW BYTES)	
#306 - #320	Array Pointers (HIGH BYTES)	
#321 - #33B	Variables @-Z (LOW BYTES)	
#33C - #356	Variables @-Z	
#357 - #371	Variables @-Z	
#372 - #38C	Variables @-Z (HIGH BYTES)	
#38D - #3C0	Labels: #38D,#38E point to label a, etc.	
#3C1 - #3C4	Used by Assembler	
#3C5 - #3FD	Free	
#3FE - #3FF	Address of plotting routines: Mode 0 set to #FE62 Mode 1 set to #F73B Mode 2 set to #F754 Mode 3 set to #F76D Mode 4 set to #F7AA	

5. DOS Workspace

#2000-#200F	Start of Disk's title
#2100-#2104	End of Disk's title
#2008-#200E	Filename of first file on disk
#200F	Qualifier of first file on disk
#2108,#2109	Start address of first file
#210A,#210B	Execution address of first file
#210C,#210D,#210E (most sig. digit)	Length of first file
#210E (least sig. digit),#210F	Start sector of first file
	(for remaining files 2-31 the same sequence is used)
#2105	Points to first unused record space in cat. buffer ie. if ?#2105=#88, there would be no files listed past #2088
#2106,#2107	No. of available sectors on the disk (normally is set to #190 (400))
#2200-#27FF	Used to store arguements for sequential files. (also used as a file buffer)

DISK PACK

Yes! Here is a long-awaited reivew of the Atom Disk Pack, written in English, yet. Not like Acorn's manuals.

The Disk Pack consists of three items: the drive unit itself, a floppy disk, and a (ahem) manual of sorts. The drive unit is housed in a really mean-looking 14 gauge aluminum case, which is considerably larger than a standard disk drive (it measures 0.30 m deep x 0.15 m wide x 0.16 m high). The reason for this is, inside the case, resides not only the disk drive but the Floppy Disk Controller (FDC) card and a heavy-duty power supply. The disk pack as it emanates from Acorn's factory does not come with the Atom bus buffer ICs (IC2 - IC5) or PL6, but some dealers might stuff them into the box anyways. It would be wise to check this out if you purchase the Disk Pack, as you won't be able to connect it without these parts. In addition to these parts, a few modifications have to be made to the Atom board. These are easy to do, and are very well described in the manual.

In addition to the Disk Operating System (DOS) on the FDC card, some additional RAM is provided. Some of this is eaten up by the DOS, but user memory is available from #3C00-#3FFF, thus giving the user a little more to play with.

The DOS itself provides the user with two different sets of new instructions. The BASIC commands, namely:

```
LOAD"<filename>"                SAVE"<filename>"
FIN"<filename>"                 FOUT"<filename>"
PUT A,W                         GET A
BPUT A,B                        BGET A
SPUT A,S                        SGET A,S
FPUT A,%F                       FGET A
PTR A                           EXT A
SHUT A
```

are essentially the same as their COS counterparts. However, in the DOS the PTR, EXT, and SHUT commands actually perform useful functions. Also, the sequential file functions are much more versatile in the DOS than in the COS. However, the details of using these are not discussed in the manual, as it says "go take a look at AT&P." However, AT&P doesn't say anything either, so......

The other set of commands are the direct DOS commands. They are as follows:

*DOS	Enter the Disk Operating System
*CAT	Same function as in the COS
*DIR	Same as *CAT except the results aren't displayed
*TITLE	Used to 'title' a disk
*SET	Used to specify the file 'qualifier'
*USE	Temporary use of the above. More on this later
*LOCK	Used to lock a file so that it can't be altered
*UNLOCK	The reverse of te above
*INFO	More like *CAT on the COS, in that it gives the Start and length of a file
*MON	Turns on DOS messages (like the *INFO messages)

*NOMON Which turns 'em off
*SAVE Identical to COS, except the qualifier is used.
*LOAD Ditto.
*DELETE Wipes out the specified file (permanently).
*GO Same as LINK from BASIC. Can be used from within Word-Pack
*RUN Like in the COS, with a few bells & whistles.
*SPOOL Everything PRINTed is sent to the disk as well.
*EXEC Executes the named text file as if it had been entered at the keyboard; eg a list of commands put on disk as a Word-Pack file
*DRIVEn Used to select the drive. (n=0 to 3)

The floppy disk that is included with the unit contains some very useful utility programs. these are FORMAT, which is used to format (initialize) disks for use; INFALL, which does a *INFO on all the files on the disk; VERIFY, which checks to see if everything on the disk is O.K.; and COMPACT, which is used to tidy the disk up somewhat. These utilities are very useful, and are not always present with some systems.

The capabilities of the Disk Pack, the manual notwithstanding, can really enhance the Atom. Each disk (single sided, single density) can hold up to 93K bytes of data. As mentioned before, each file can have its own qualifier, which is like a 'key', in that the right qualifier must be specified in order to access the file. Hence, it is possible to have a number of files with the same name but with a different qualifier. Another useful bit is the ability to give a disk a title, which is displayed whenever the *CAT command is used. Some unfortunate drawbacks include a limit of only 31 files, and a limit of only 7 characters in filenames. Thirteen character filenames in the COS were bad enough, but this is clearly ridiculous. It seems that the reason why these restrictions have been imposed is the size of the catalogue buffer. Why they just didn't have a bigger buffer is beyond me.

As mentioned before, another serious hindrance is the abysmal "manual". Being only 20 pages long, including covers, it is most likely the worst manual I've seen for anything. Written in the true style of Acornsoft (no mention is given of the author's name presumably out of fear of having him lynched) it seems to out-dismal even the BBC board manual (which I don't have). One very weird thing about the manual is that it gives a very complete description of the hardware, including complete schematics and parts lists, as if they were almost inviting enterprising beings to go out and clone the darn thing!

Probably the most noticeable advantage of the Disk Pack is its rapid speed. Whereas the COS will take 10 minutes to load a 10K program, the Disk Pack can do it in seconds. This makes it much easier to write and debug software.

One last thing, though. Since the internal power supply is used to supply power to the Atom (the +8V supply is no longer needed) this means that any memory boards inside the Atom that need an 8V supply will have to be connected to the internals of the power supply. This means having to trail a wire out of the Atom to the Disk Pack and is definitely uncool.

by: Colin Hinz

To B.B.C. or....
Not Two B.B.C.s

That is the question (or questions). Whether it is nobler of the mind to suffer the slings and arrows of outrageous fortune or to pension off the Atom and trade up to a B.B.C. microcomputer. It is your choice. The B.B.C. is here, available in Canada and ready to run on 110V 60Hz current. A North American version of the machine is being distributed in Canada by E.M.J. Data Systems of Guelph, Ontario, who very kindly made a computer and disk drive available for this review.

Although the B.B.C. is by Acorn, and is certainly an outgrowth from the Atom, it is a far cry from the little old machine that we know so well. For a start, it comes with documentation, two thick, high quality, three ring binders of it. Then, it comes fully equipped and ready to go. The computer as sold includes a switching power supply and interfaces for a monitor or TV, RS423 (an extension of RS-232C), cassette, analog input (joy-stick?), Econet local area network, disk drive, printer, user I/O, plus the bus and the "Tube". The recommended disk drives are dual, double sided, 80 track drives with their own power supply. These give a total of 800 K of on-line storage, but single sided Olivetti or Shugart 40 track drives work fine.

As it uses the 6502 CPU, it is essentially a 64K machine, but this is used effectively and it includes a 16K block which is paged ROM. Three pagable 16K ROMs are installed, BASIC, DOS and VIEW (a word processor). The operating system, (in a fourth 16K ROM), selects the required paged ROM in a manner transparent to the the user. The machine powers up in BASIC and the disk operating system, which then searches drive 0 for an optional BOOT file. Word processing is selected by *WORD. To get back to BASIC simply enter *BASIC. Of the 32K of RAM almost 7K is reserved for operating system use and from 1K to 16K is used for the screen display.

The screen display warrants a review all to itself. There are eight different modes. You can select 20, 40, or 80 columns in 2, 4, or 8 colour display with each colour being available steady or flashing. There can be 20, 22, or 25 rows. Most modes allow text and graphics to be mixed. Optionally, windows can be defined for the text and the graphics. Then the graphics show over only part of the screen, whilst the text (eg. program) is shown and scrolls by itself in another part. The graphics resolution ranges from 160X200 to 640X200. Obviously, higher resolution, more columns and more colours use up more memory. But! if the RAM is not being used for the screen then it is available in a contiguous block for your program.

The BASIC is superlative. It refines the idiosyncracies of the Atom, civilises them, then expands the language to include many of the things which were missing on 'old faithful', plus others which most of us havn't even dreamed of. It allows functions to be defined then called by name. Similarly with procedures, which are like sub-routines with a proper name and their own use of variables. The variables can optionally be common to the main program or can be completely separate. Or, can be passed to the procedure changing their name as they are passed. Variable names, incidentally can be of any length, with all characters recognised. There is IF...THEN...ELSE, which together with the Atom style REPEAT....UNTIL, all adds up to a BASIC which allows very structured programming. Other niceties are Microsoft style string handling, with LEFT$, MID$, and RIGHT$, also INKEY, INKEY$, READ...DATA...RESTORE, and good support for the graphics capabilities including a PLOT command which allows the painting of a defined area. The area must be a triangle but it makes up for this shortcoming by being phenomenally fast. There is a SOUND command which is used to control the four sound channels and the built in speech processor. Yes! the thing will even talk to you. These notes will serve only to highlight some of the nice features of BBC BASIC. To give some idea of the overall scope, over 120 keywords are recognised by the BBC compared with less than 50 on the Atom. It is also very fast, not only because of the 2 MHz 6502, but mainly because of the firmware.

The disk filing system is similar to that on the Atom, with a few changes. There are new commands including *COPY and *BACKUP, which do a filcopy and a discopy, *DUMP and *TYPE which do a hex or an ASCII listing of a disk file, and the commands to select the BOOT procedure. This allows 4 options. The file !BOOT on a disk can be: 1) ignored, 2) *LOADED, 3) *RUN, or 4) *EXECuted. This command is stored on the disk and is effective on power-up. Again a major difference lies in the documentation. There are nearly ninety pages on the disk filing system, starting from the most simple concepts up to technical data on file storage addresses and vectors. Each command is explained by it's purpose, examples of use, a description of what it does, a list of associated commands and notes if required.

The word processor VIEW is another comprehensive piece of firmware, with many similarities to WordStar. The text can be entered using the 34 column screen, corrected with direct on screen editing. Each paragraph can be formatted using a "ruler" which incorporates margins and TAB stops.The standard page format has a top margin, a header line, and a header margin followed by the text. Then, a footer margin, a footer line, and a bottom margin. These headers and footers will print out on each page but it is up to you what, if anything, is written on them, and the number of blank lines can be changed as required. Control codes can be embedded anywhere in the text. These operate through a printer driver so that, in principle, the same text file could be used with various printers to give underlining or bold type as required, provided that the appropriate driver was loaded. Unfortunately the default driver in VIEW had no effect on a Centronics 739 and no appropriate driver routine was available.

The file may then be previewed with the SCREEN command after setting the mode to 80 column. This displays the text as it will be formatted to the printer showing the effect of centering text and showing page breaks. The individual line breaks are shown all the time in the normal editing and entry mode. Documents may be printed out from memory or from disk to printer. This allows files to be much longer than would fit into memory.

One thing is becoming apparent. This computer will be much better supported than the Atom ever was. It is being aimed at the educational market though it will certainly find much wider application. Several packages of software were provided for evaluation. These ranged from elementary school execises to recognise the shape of words to grade 8 or 9 math and science lessons. The lessons are well presented and have been completely "translated" from English to American. Teachers who have seen them felt that they were better than most material presently available and included the little extras they were looking for when evaluating software. The general format was instruction, followed by some form of test and a game based on the subject matter. In many cases the program would keep a record of each childs score in the test for use by the teacher. The use of auto-booting disks and the facility of the BBC to allow redefinition of keys leads to programs which were very difficult to crash. Even pushing the BREAK key simply led to a beep with instructions on the screen showing how to continue.

It was mentioned earlier that this is a North American version of the BBC Microcomputer. There are some subtle changes from the British equipment. The hardware has been changed to suit North American colour television, and the screen output changed to allow for the lower resolution. Specifically there are fewer lines of text allowed, though the number of columns remains unchanged. It must be pointed out at this time that despite all the instructions, the computer provided for evaluation could not be made to run on a colour TV, and the distributer was not able to demonstrate this facility. It was evaluated using a monitor which is how it will probably be most often used. We were not able to check how software written for one version of the BBC would run on the other. If the format of the screen display were important to the running of the program there would probably be problems. If you wish to run the British version here one or two small changes are required. A very simple jumper change is needed in the power supply and you may need to change the vertical hold circuit of a monitor to accomodate the 50 cycle scanning. This latter change may be as simple as altering the control or could involve the addition of a capacitor to the control circuit. One member of the User' Group who has got himself a British BBC found that he had to add the capacitor but having done this the monitor works equally well, without adjustment on his Atom and the BBC. You would not, of course be able to use the British version with a colour TV.

The choice mainly comes down to one of software. If you want to use the excellant educational software available for North American use, then you will probably want the North American Version. If you will be writing your own software or want to use the extensive software available in Britain, then you may be prepared to go through all the hassle of importing a British version.

Overall, a fine machine worthy of its heritage. If it takes off in Canada as it deserves to, we may have to change our name to the Acorn Users Group to cover both computers.

RAM into ROM

If you have a disk drive you may have wished that you had RAM in block #A000. Then you could load WORDPACK or TOOLBOX from disk, and avoid having to change the EPROM.

Some people of course have a ROM board which allows the ROMs to be selected through software. Here is another way for those who have the 16/32 K RAM board replacing the normal lower text space RAM from #2800 to #3C00.

If you had to remove IC10 to IC19 when fitting your RAM expansion board, then you can use some of these sockets as block #A000 by slight changes to the chip select lines.

First remove IC6 (a 74LS138) and get a 16 pin low profile DIP socket. Bend out pins 3, 4, 7, 12, 13, 14 and 15 on IC6 and insert the remaining pins in the socket. Then jumper pin 3 on the chip to pin 4 on the socket and pin 4 on the chip to pin 3 on the socket. Also jumper pin 7 on the chip to pin 12 on the socket. Finally re-insert the chip/socket assembly in the original IC6 socket.

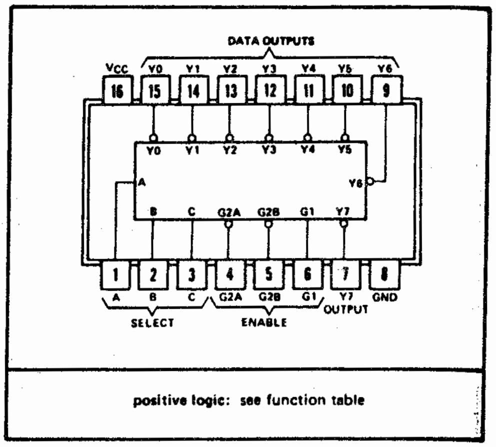
This sounds complicated but its easier than it sounds. The jumpers can be easily made with a small hand wire wrapping tool, or by careful soldering. Switching the inputs to pins 3 and 4 causes the required chip select signals to appear on pins 7, 9, 10 and 11. Jumpering pin 7 on the LS138 to pin 12 on the socket allows IC sockets 12 to 19 to be used for block #A000 by putting back eight of the 2114 chips which were obsoleted by the memory expansion board.

If you take this route, remember to save your utility ROMs on disk before making the change. Each one should be *SAVEed from #A000 TO #B000 with the appropriate execution address. Eg.

```
          *SAVE WORDPAK A000 B000 ACFD
```

The execution addresses for CALC and T/BOX are #A093 and #AF00 respectively.

SN54LS138, SN54S138 . . . J OR W PACKAGE
SN74LS138, SN74S138 . . . J OR N PACKAGE
(TOP VIEW)



positive logic: see function table

'LS138, 'S138
FUNCTION TABLE

INPUTS					OUTPUTS							
ENABLE		SELECT										
G1	G2*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

*G2 = G2A + G2B
H = high level, L = low level, X = irrelevant

WORDPAK PLUS

Wordpak Enhancement Package

Rick Bales has put together one of the most sophisticated and comprehensive software packages yet. Wordpak Plus is intended for use with the Acorn Atom equipped with Word Pack and a disk drive. It allows for the insertion of printer control codes at any place within the text. The embedded control codes appear as inverse characters and may be deleted, moved or otherwise edited just like any other characters. They do not interfere with the correlation between the cursor and the text.

When the text is sent to a printer Wordpak Plus interprets the control codes and sends the appropriate string of codes. Each embedded code may represent up to three printer control codes. So far versions have been compiled for the Gemini 10X, and Centronics 737/739 printers. Versions for other printers may be compiled if necessary.

Wordpak Plus uses its own set of embedded codes. It is possible that text with embedded codes could be used with different printers, version of driver was loaded. To this end an attempt is being made to ensure that at least a subset of available printer commands use the same control codes in the separate versions compiled. These would be

1. Underline
2. Condensed
3. Expanded text
4. Reverse half line or superscript
5. Forward half line or subscript.

Included in the package is a HELP menu which may be called up from within Word Pack to remind you of the codes for various printer commands.

Associated with the printer enhancement package is a set of utilities which can all be used from within Word Pack without corrupting the text.

WORDPAK PLUS UTILITIES

The WORDPAK PLUS UTILITIES are a group of routines designed for use with Wordpak, with or without the use of the Wordpak Plus printer enhancement package. The utilities consist of the following:

1. <u>RECOVER</u> - recovers text lost after a BREAK

2. <u>APPEND</u> - appends a disk file to the end of the current text

3. LEARN - "teaches" Wordpak a number of keystrokes that are then saved as a disk file...sort of a disk-based function key routine

4. GET - retrieves a LEARNed file from disk

5. BUFFOUT - saves Wordpak's buffer to disk

6. BUFFIN - loads Wordpak's buffer with the contents of a disk buffer file

7. UPDATE - an alternative to Wordpak's SAVE routine that automatically creates a backup file while saving the current text

In addition to the above, two general purpose routines are included that can be used on any disk files, programs or text, namely:

8. LOKALL - locks all files on a disk

9. OPENALL - unlocks all files on a disk

10. RENAME - renames any file on the disk under the current qualifier

I trust that by now you are saying "Wow that sounds pretty good". Well the best is yet to come. Rick has made the whole lot available to members of the Users' Group. If you send in a formatted disk the programs are yours free of charge. Alternatively send five dollars to cover the cost of the disk and packaging. Please specify your printer when writing. Sorry but it is not feasible to offer this service on cassette. The utilities, of course, can only work with a disk anyway.

Now, if only someone will write an output routine to maintain the right justification for text incorporating printer control codes...... And while you are at it, how about a printer driver to use the micro-justification available in the proportional spacing type face on a Centronics printer.

MULTIPLE DRIVES ON THE ATOM

Adding an extra single or double sided drive to the Atom is a straightforward matter. It is simply connected in parallel to the existing drive, using a longer length of 34 way flat cable. If you are careful you can reuse the existing connectors from the cable inside the Atom Disc Pack. Add an extra edge card connector, run +5V, +12V and GND to your new drive and Voila! Each drive has jumpers which are used to designate it as drive 0 or drive 1. If you fit a double sided drive, the two surfaces are considered as seperate drives, either drive 0 and drive 2, or drive 1 and drive 3.

LETTERING IN GRAPHICS 4:

by John Lasruk

And I know just what you're going to think when you start reading this: "Oh, GAWD! Another Wordpack article." This one is a lulu, though, because it makes that little ROM do a merry wee jig. I have to say, before we get too far into this thing, that I was informed, after I had "discovered" this trick, that someone else had it first. I don't know who; it apparently appeared in Acorn User sometime. Breaks my heart; now I know how the third person to discover photography might have felt when he found out...

If you own a SOFT VDU tape, you have what you need to render little typey things unto your graphics 4 art. Well, everything EXCEPT the 1.5K of memory the so-and-so eats up. But your Wordpack (assuming you own one) has its own built-in graphics 4 character generator. All we have to do is access it. Easy enough.

Atom normally jumps to #FE94 to read the keyboard and #FE52 to take what it just read and make some sense of it on the VDU screen. Actually, the stupid little idiot has to look up these places in his "address book", in a process called "indirection". The "address book" lives in RAM, so whatever is written there can be rubbed out and new things written in. When you call up the Wordpack, it places the address #ACCE in #208, #209 and #AC7F goes to #20A and #20B. These are respectively Wordpack's write character and read character routines.

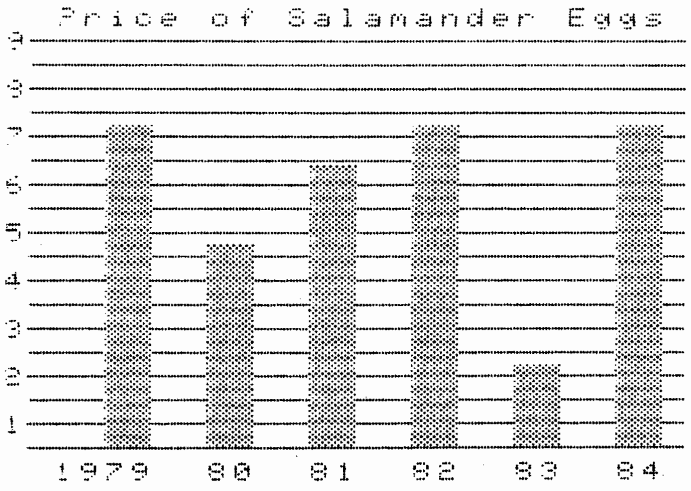
All we have to do is zip the right four byte word to #208 and we can use the Wordpack's very own proprietary character generator for our programs. The accompanying bit of fluff does this automatically and switches back again, but perhaps you'd like to use the character generator in direct mode. Here's how:

First enter: !#208=#AC7FACCE <RETURN>. Now enter <CTRL><L> and <CTRL><X>. There you are! Isn't it wonderful? Go ahead... fondle the screen. It's warm... it's soft... it's... it's... not all that easy to use. You can write and list a program, but screen editing is a problem, because <COPY> isn't recognized. The cursor control keys seem to work beautifully -AUTO REPEAT!!!- but they repeat only 64 times (one full line buffer, methinks). Then, THEN, one presses the return key, um, like this... BEEP! ERROR 94. Why, you dirty little son of a... Ahem. See what I mean.

But, hey! You can still use this trick to letter charts and other fabulous graphical computer dickey doos by calling the generator up from your BASIC programs. And you do get to keep all of your lower text space memory, hmm? It might be of some value for you to know that Wordpack's read character routine is not, strictly speaking, necessary, so setting !#208=#FE94ACCE will give you the graphics 4 lettering using the regular read character routine. By the way, you aren't actually setting the word processor in motion, but Wordpack has to be in a position to be read, so other utility ROMs can't be used. This may be a problem if you are really into Toolbox ROM commands. Forget them. Now, don't cry; life is sometimes brutal that way. Say! I nearly forgot! Here's a program for you. You deserve it.

Important information Chart:

```
100 !#208=#AC7FACCE                    ***set the vectors, Mr Spock!
110 PRINT$12';?#E1=0;@=0
120 CLEAR4
130 F.X=9 TO 1 STEP-1                  ***print chart numbers
140 P."  ",X''
150 NEXT
160 PRINT"    1979  80  81  82  83  84"
170 PRINT$30"    Price of salamander eggs"
180 FOR Y=25 TO 180 STEP 9             ***draw horizontal lines
190 MOVE 25,Y;DRAW 230,Y
200 NEXT
210 FOR A=49 TO 209 STEP32             ***draw bars
215 Y=A.R.%10+1
220 FOR X=A TO A+14 STEP 2
230 MOVE X,25; DRAW X,((15*Y)+25)
240 NEXT
250 NEXT
260 !#208=#FE94FE52                    ***return to Earth, Scotty!
270 FOR S=#8000 TO #97FF STEP 4        ***reverse screen BASICally
280 !S=!S:-1;NEXT
290 LINK#FFE3                          ***wait for a keypress and die
300 END
```



NEW ROM INFO

Michel Charest, 1640 Fleury E. #3, Montreal, H2C 1S8, has provided information on a utility ROM giving new SCREEN control, including 32 or 42 characters per line, windows, and mixed text and graphics. Also on the DISATOM SUPER ROM another "toolbox" type ROM with several commands related to loading from cassette. These include HIGH (1200 baud), LOW (300 baud), HELP and TAPE to recover really bad tapes.

News from Local Groups

Several local groups are active here is news of some of them.

The Halton Acorn Users' Group usually meets in Burlington, ONT. but may be moving towards Oakville. The next meeting will be on Tuesday, October 16th at Saint Marks School, on Upper Middle Road in Burlington at 7:30 pm. Come along and help formulate plans for the coming year.

The Toronto Hardware Development Group meet in the North of Metro at 7:00 pm on the second Wednesday of each month. Contact Dean Gunby or John Chiappetta on 630-3776 during business hours.

The TransAlta Group would like to hear from others in the Calgary area. Contact Jack Ladick 267-3850 (Bus) or 256-3983 (Home).

FOR SALE

TIMEDATA MZ165 ROAM BOARD. CONTAINS 4K OF CMOS RAM, PLUS WORDPACK, ATOMCALC, AND PROGRAMMER'S TOOLBOX ROMS. ALL MANUALS. BEST OFFER. CONTACT MITCH BAKER, 307 - 50 KING ST. E., DUNDAS, ONT. L9H 5NH.

ATOM: 6K RAM - 12K ROM - TOOLBOX - PWR SUPPLY - JOYSTICK - OVER 25 GAMES - BOOKS - MAGAZINES - $159.00 --ATOM 32K EXPANSION BOARD - $299.00 -- COMPUTER AND RAM BOARD FOR $399.00
Trevor C. Sadler, 4 Croxton Rd. E. London Ont. (519) 438-8932

Acorn Atom - 12K RAM + 12K ROM, Toolbox, printer interface, external bus extension, software (incl. Forth) AND Centronics 737 Printerand Cables -- $400. Steve Gibson, 1121 Spears Road, Fort Erie, Ont., L2a 4N2. (416) 871-3265 after 6 pm.

Colin Hinz 1118 College Drive. Saskatoon, SASK. S7N 0W2 has a wide range of chips and other hardware componants for sale at low prices. Write to him for a list.

Atom with disk drive and software. Best offer. Milan Vojnic
231 6th Ave W. Owen Sound, ONT. N4K 6J2 Tel. (519) 376-7864

Fred Springer knows of two people trying to sell their Atoms. One with disk drive one with Centronics printer. Call or write for details.905 Glenwood, Burlington, ONT. L7T 2J8 (416) 632-2044

The Users' Group has been offered a batch of colour boards, at a price of $30 each, if we take the lot. Is anyone interested?

EDG Electronic Distributers advise us that they are no longer supporting the Atom although Gladstone retail have some items at clearance prices.

Much of the English software for the Atom is now available through SOFTWARE CLASSICS, 2, Connie St., Openshaw, Manchester M11 2JD, England. They also have various EPROMs. All the cassette based programs and ROM firmware is also available on disk.